1Frame2Work

Web service client for iOS

Version 1.1.4

USER MANUAL

Content

Idea behind 12FrameWork

I started this app two years ago along with learning and mastering Objective C and Cocoa Touch platform. I also have more than 15 years of experience in development of business application for Microsoft Windows **™**.

Main idea behind 12FrameWork is, to somehow connect that completely different platforms or more precisely, different worlds.

How it begins? My collegue works as a safety engineering manager for known European oil company and gets iPad from company. She asked me: »What useful for my job I can do with iPad/iPhone, except surfing, reading mail and playing games?« She travel a lot to company affiliates and she would like to have some useful schedule lists (directly from company databases), checklists for safety control, easy way to connect iPad images she made with corporate data and at last, to enter useful data directly to company database with iPad. Web browser apps are slow, far from comfortable and most important, not native (cannot use camera, location services etc). Developing native app is almost impossible, because IT (Information Technology) sector of company do not develop for iOS devices. Why? It is not cost efective to hire iOS developer(s), purchase enterprise licence, and all that stuff only for couple of users.

On contrary, any existing IT can quickly make some useful web services and expose database tables to iOS users.

They only need appropriate web service client on iOS devices. And that's where 12FrameWork jumps in.

So I decided to make this app with my friends and collegues, as universal, customisable, useful that will grow in features and reliability, with every new version and with every new user.

Thank you for using 12FrameWork
Milenko, Taya&Niny team

Purpose, description and possible use cases

12Framework is a universal web service client for iOS, developed in Objective C.

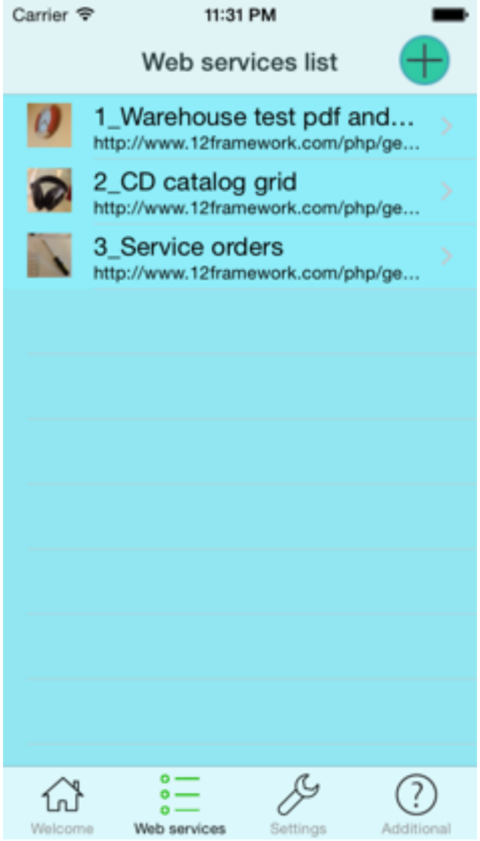
It can be used for:

- Presenting data from web services in DataView, GridView or Text debug mode, with custom request parameters
- Appending, updating and deleting data on web services in custom dynamic forms, supporting text, bool, date, multiline text, pdf links, barcode reader and imagepicker controls.
- Shooting images and connecting them with database records
- Testing web services

12FrameWork offers easy export/import of webservices settings via email (also import settings from http location), protected by custom passcode.

Application is free to download and use. Adding new records and displaying more than 20 rows, are avaliable as a single InApp purchase.

Adding new web service



Carrier 11:31 PM

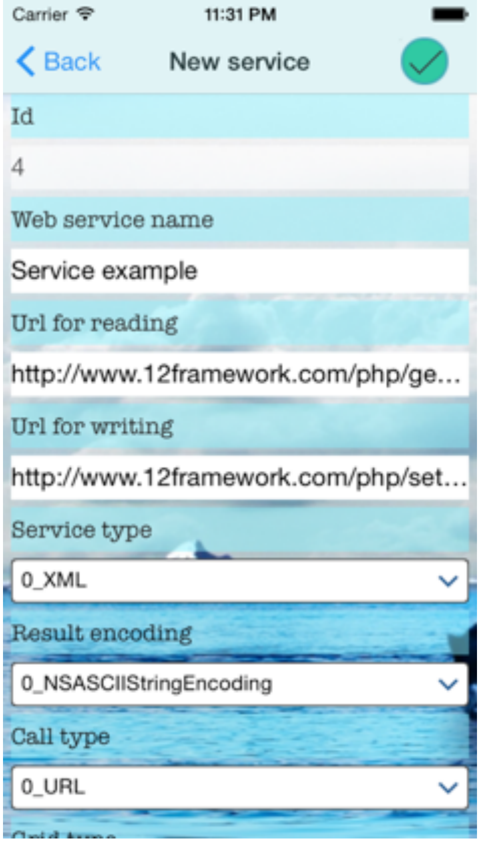
Web services list

1_Warehouse test pdf and...
http://www.12framework.com/php/ge...

2_CD catalog grid
http://www.12framework.com/php/ge...

3_Service orders
http://www.12framework.com/php/ge...

12framework provides possibility to specify and call unlimited number of web services.



Carrier 11:31 PM

Id: 4

Web service name: Service example

Url for reading: http://www.12framework.com/php/ge...

Url for writing: http://www.12framework.com/php/set...

Service type: 0_XML

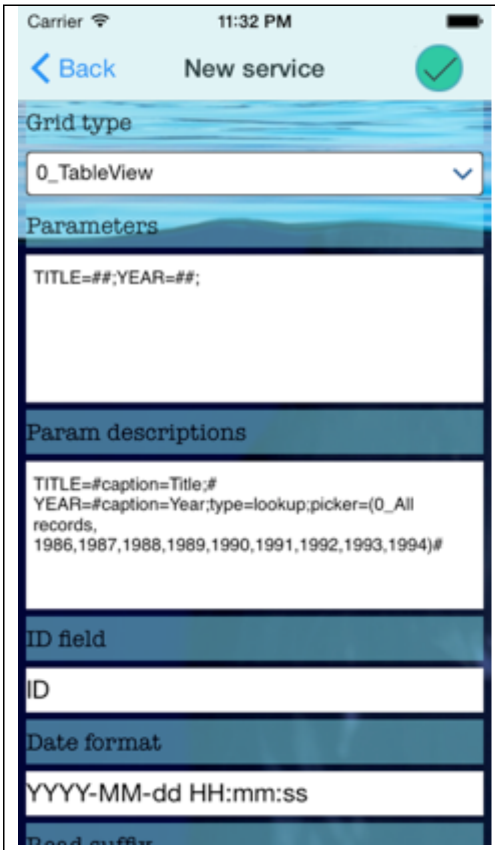
Result encoding: 0_NSASCIIStringEncoding

Call type: 0_URL

Click on + brings in input form for entering web service data, prefilled with some custom data (**WARNING: web service settings data are case sensitive**):

- **Id** is auto increment identificator for web service and cannot be changed
- **Web service name** is user define custom name of the web service
- **Url for reading** represents url to get data from server
- **Url for writing** represents url to post/write data to server
- **Service type** can be XML or JSON. XML offers sorted field order (dynamic form fields will be created in the same order as a table fields on database server), on the other side, JSON has unsorted field order
- **Result encoding**, encodes web service result to appropriate encoding.

Call type, it can be URL or HTTP post. URL means that all parameters are nested in http url line. With HTTP post, you can send parameters as custom http body.



Grid type
0_TableView

Parameters
TITLE=##;YEAR=##;

Param descriptions
TITLE=#caption=Title;#
YEAR=#caption=Year;type=lookup;picker=(0_All records,
1986,1987,1988,1989,1990,1991,1992,1993,1994)#

ID field
ID

Date format
YYYY-MM-dd HH:mm:ss

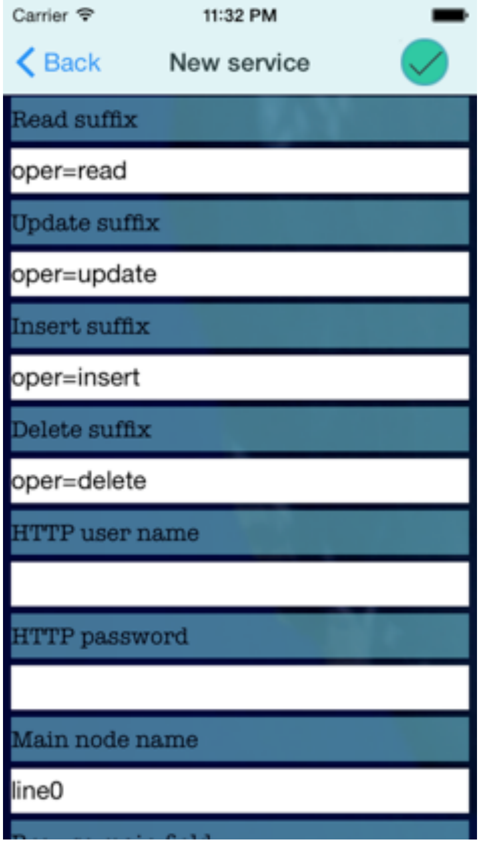
Grid type can be:

- **TableView**, native iOS tableview control.
- **GridView**, experimental gridview component for displaying all fields data in a grid.

Parameters and param descriptions:
Define params for web service call. According to presented example, it's easy to define unlimited custom textedit, barcode, lookup or date field params for calling web service.

ID field:
Enter unique key field name for web service dataset. It is mandatory setting, otherwise insert, update and delete of particular record will not work.

Date format:
Enter date format compatible with your web service response



Read suffix
oper=read

Update suffix
oper=update

Insert suffix
oper=insert

Delete suffix
oper=delete

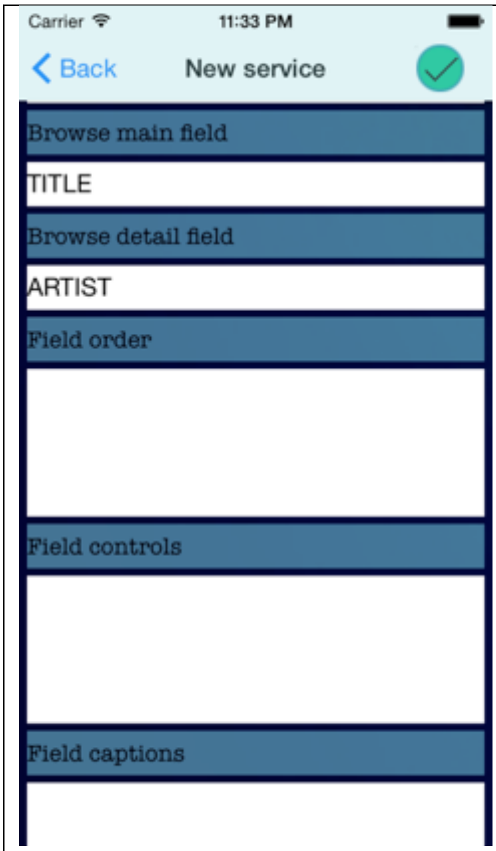
HTTP user name

HTTP password

Main node name
line0

- **Read suffix**, suffix which will be added to »Url for reading«, when you get data from server, applies only for URL call type
- **Update suffix**, suffix which will be added to »Url for reading«, when you post updated data to server, applies only for URL call type
- **Insert suffix**, suffix which will be added to »Url for reading«, when you post inserted data to server, applies only for URL call type
- **Delete suffix**, suffix which will be added to »Url for reading«, when you post delete request to server, applies for URL call type only
- **HTTP user name**, user name for login on web server
- **HTTP password**, password for login on web server

Main node name, node in XML or JSON result which identifies one record from web service response



Browse main field

TITLE

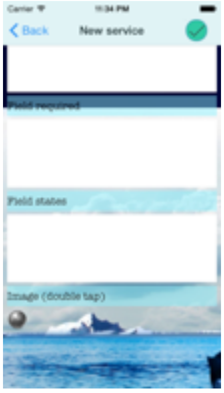
Browse detail field

ARTIST

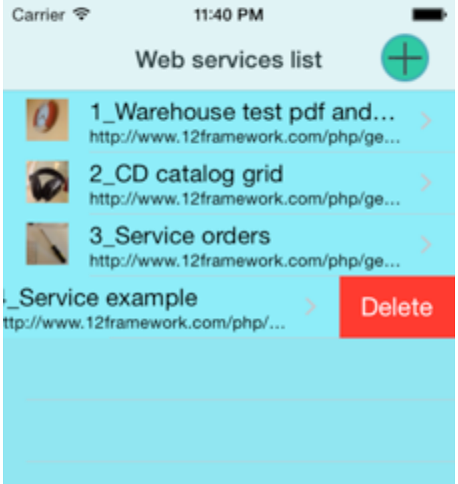
Field order

Field controls

Field captions



- **Browse main field**, field in XML or JSON result which will appear as a first label on TableView result
- **Browse detail field**, field in XML or JSON result which will appear as a second label on TableView result
- **Field captions**, after first call of web service, captions are filled with database field names and can be simply changed respectively
- **Field controls**, after first call of web service, control types are set as text fields. Can be changed to: **text**, **memo**, **date**, **bool**, **imagepicker, pdf** and **barcode** (be careful of case sensitivity). Text is one row text component, memo is multirow text component, date invokes datepicker field, bool invokes switch for boolean fields (saves 1 for True and 0 for False to database field). Imagepicker is imageview field which displays image shooted on device camera according to web service result field. It saves image automatically to local application documents\image\service_id directory and posts image name to assigned database field. Camera invokes with double tap. Image name is formed with following pattern row_id_12frmYYYYMMDDHHMMSS.jpg
- **Image**, double tap and shoot appropriate image for your web service. It will then appear in web service list
- **PDF**, field must return HTTP link value, which creates or responses with PDF file
- **BARCODE**, barcode field triggers camera which scans barcode and write it in your service editbox.



1_Warehouse test pdf and...
http://www.12framework.com/php/ge...

2_CD catalog grid
http://www.12framework.com/php/ge...

3_Service orders
http://www.12framework.com/php/ge...

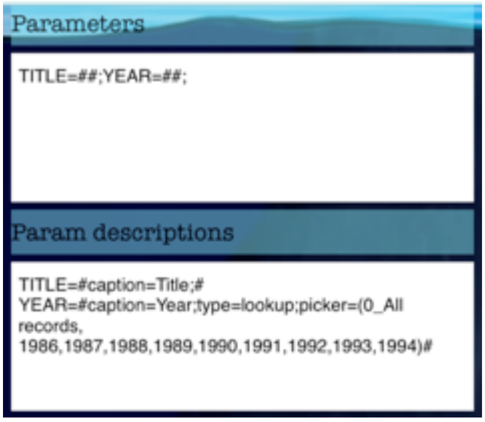
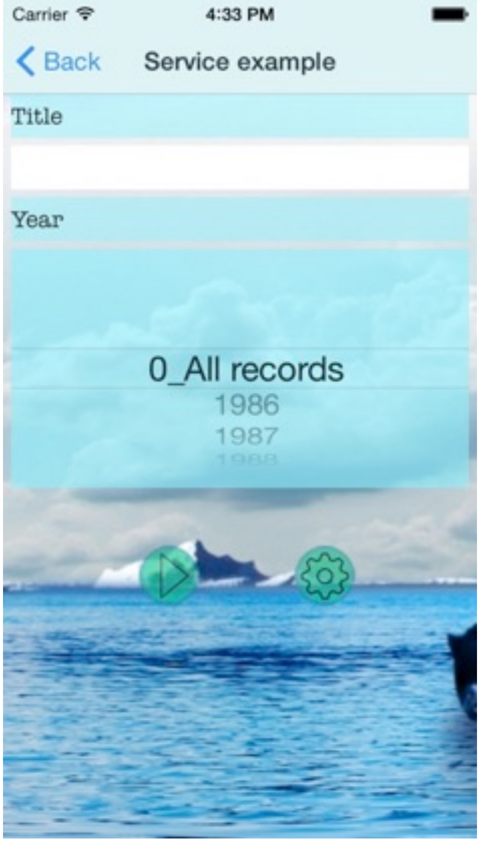
_Service example
ttp://www.12framework.com/php/...

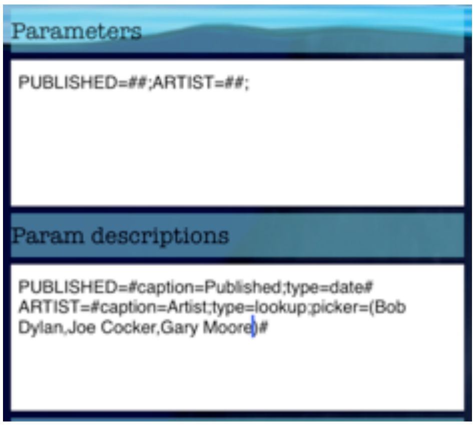
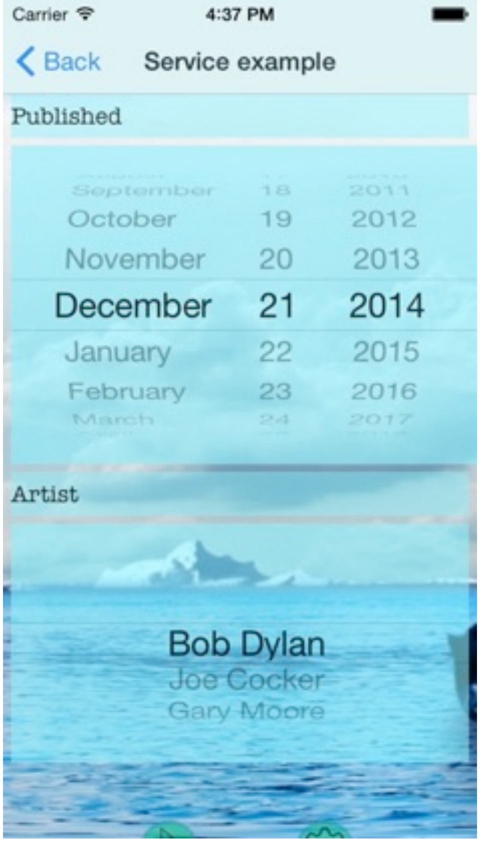
Deleting web service

Simply swipe to left selected web service and tap on delete. It will immediately delete service without further warning.

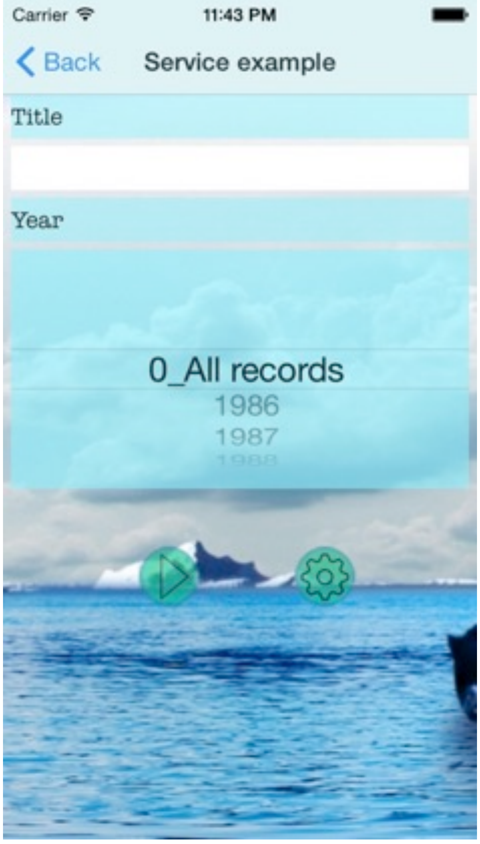
Setting parameters for calling web service

Examples: If you need static table with key value pair, declare list with key_value, for example: 1000_New York,2000_London,3000_Tokyo,4000_Moscow,5000_Melbourne...

 Parameters TITLE=##;YEAR=##; Param descriptions TITLE=#caption=Title;# YEAR=#caption=Year;type=lookup;picker=(0_All records, 1986,1987,1988,1989,1990,1991,1992,1993,1994)#	 Title Year 0_All records 1986 1987 1988

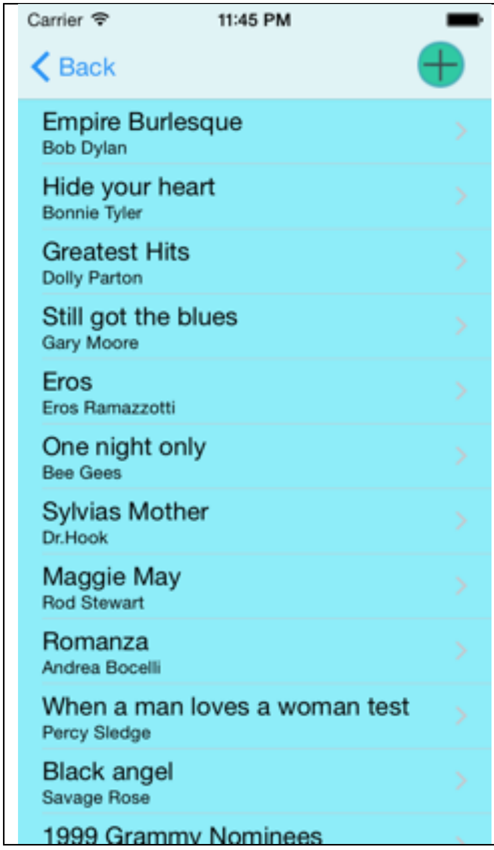

Parameters
PUBLISHED=##;ARTIST=##;
Param descriptions
PUBLISHED=#caption=Published;type=date#
ARTIST=#caption=Artist;type=lookup;picker=(Bob Dylan,Joe Cocker,Gary Moore)#
	
	
	Published
	September 18 2011
	October 19 2012
	November 20 2013
	December 21 2014
	January 22 2015
	February 23 2016
	March 24 2017
	Artist
	Bob Dylan
	Joe Cocker
	Gary Moore

Editing web service

 Carrier 11:43 PM Back Service example Title Year 0_All records 1986 1987 1988	If you want to change entries of web service, tap on gear button ⚙ After changing, save it with click on ✓

Starting web service

Web service call starts with tap on play button ▷



Empire Burlesque
Bob Dylan

Hide your heart
Bonnie Tyler

Greatest Hits
Dolly Parton

Still got the blues
Gary Moore

Eros
Eros Ramazzotti

One night only
Bee Gees

Sylvias Mother
Dr.Hook

Maggie May
Rod Stewart

Romanza
Andrea Bocelli

When a man loves a woman test
Percy Sledge

Black angel
Savage Rose

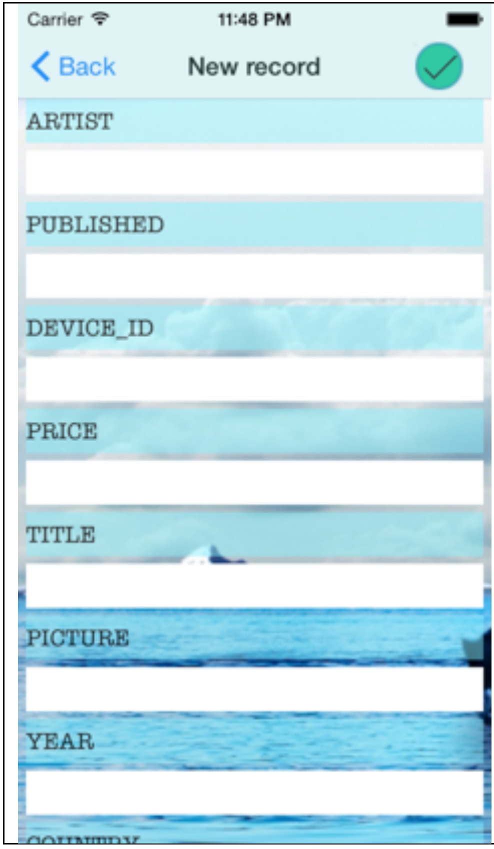
1999 Grammy Nominees

If Grid type is TableView, list of records from web service will be displayed.

WARNING: *Free version is limited to display first 20 rows only*.

First label is what was set in Browse main field
Second label is what was set in Browse detail field

Tap on + sign (*feature avaliable as InApp purchase*), you add new record...



ARTIST

PUBLISHED

DEVICE_ID

PRICE

TITLE

PICTURE

YEAR

COUNTRY

... and fill data.

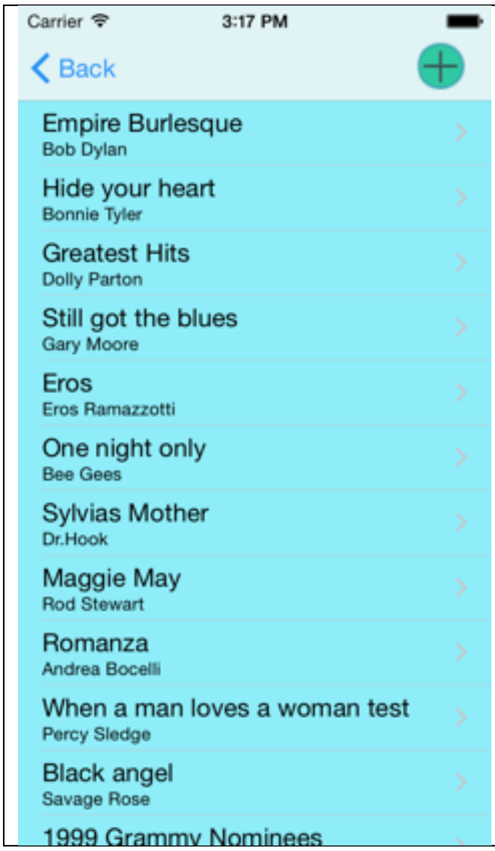
For ID field is better, to let server side to assign the value, but you can type it anyway.

After typing values for some of avaliable fields, tap on checkmark ✓ to post data to web service.

You will be returned to previous list of records, with new record at the end of the list.

TableView refreshes after posting data, so if you see your record in list, it 100% exists in the server database.

Editing web service result



Carrier 3:17 PM

Back

Empire Burlesque
Bob Dylan

Hide your heart
Bonnie Tyler

Greatest Hits
Dolly Parton

Still got the blues
Gary Moore

Eros
Eros Ramazzotti

One night only
Bee Gees

Sylvias Mother
Dr.Hook

Maggie May
Rod Stewart

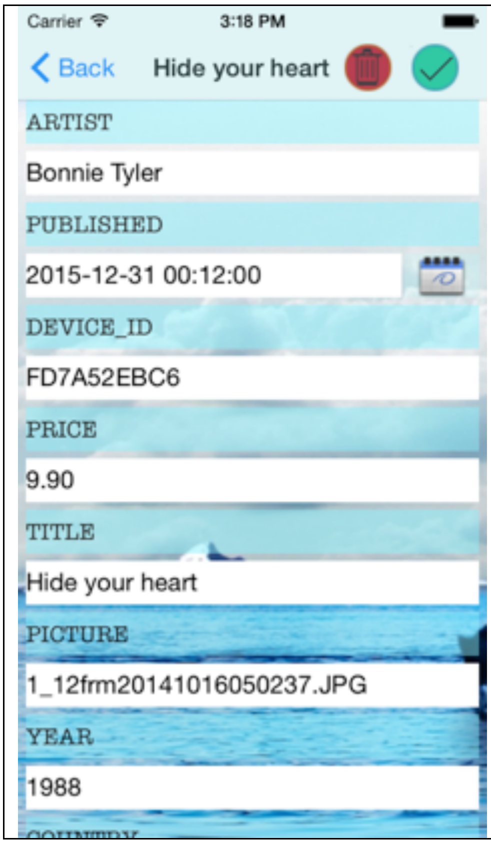
Romanza
Andrea Bocelli

When a man loves a woman test
Percy Sledge

Black angel
Savage Rose

1999 Grammy Nominees

Tap on one row...



Carrier 3:18 PM

Back Hide your heart

ARTIST
Bonnie Tyler

PUBLISHED
2015-12-31 00:12:00

DEVICE_ID
FD7A52EBC6

PRICE
9.90

TITLE
Hide your heart

PICTURE
1_12frm20141016050237.JPG

YEAR
1988

COUNTRY

... and app will display data ready for editing.

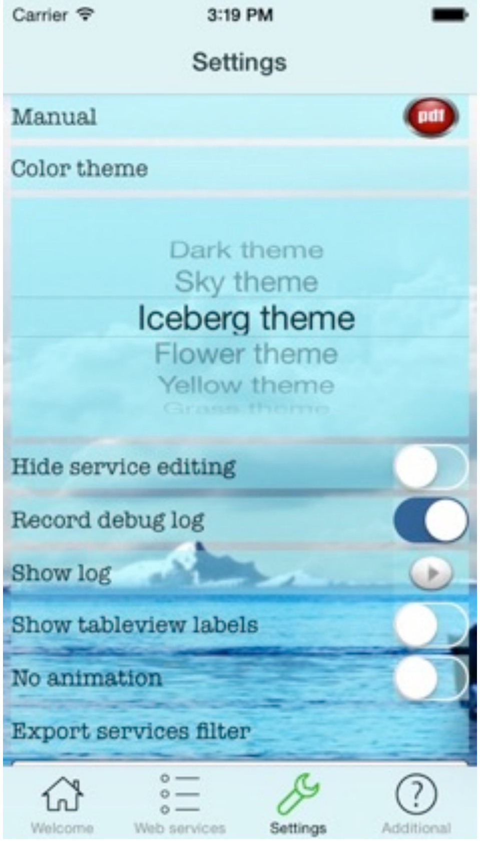
After changing values for some of avaliable fields, tap on checkmark ✓ to post data to web service.

TableView refreshes after posting data, so if you change browse main or browse detail field it will reflect changes immediately.

To delete record, tap on 🗑

After confirming, record will be deleted and TableView refreshes to reflect changes.

General settings, export, import, color themes, encoding and custom background



Carrier 3:19 PM

Settings

Manual

Color theme

Dark theme
Sky theme
Iceberg theme
Flower theme
Yellow theme
Grass theme

Hide service editing ○
Record debug log ●
Show log
Show tableview labels ○
No animation ○
Export services filter

Manual, shows this document on app internal viewer

Color theme, repaints all application views in choosed color scheme

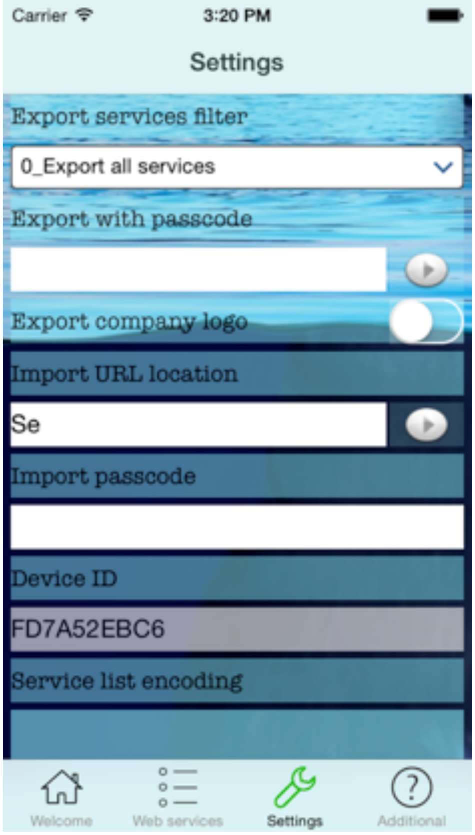
Hide service editing, hides gear ⚙ button so you cannot edit existing web service settings

Record debug log, turn on this switch to begin recording log

Show log, if you have trouble calling your web service, or you want to see raw communication between application and web service,

Show tableview labels, if turned on, tableview will display also field name with field value.

No animation, turns off animation of forms and controls



Carrier 3:20 PM

Settings

Export services filter
0_Export all services
Export with passcode
Export company logo ○
Import URL location
Se
Import passcode
Device ID
FD7A52EBC6
Service list encoding

Export services filter, when you add and set web services, you have to distribute settings to your users. Export filter enables exporting all services and one by one.

Export with passcode, sensitive web service settings data must be properly protected, thus enter custom passcode and tap on ▶. Selected services will be exported to xml and packed to ZIP with typed passcode protection. Mail composer will be opened with prepared attachement 12fwservices.fwz and appropriate text, so you can enter email addresses and send settings. The other way is to send settings to yourself, open mail on computer and rename 12fwservices.fwz to 12fwservices.zip. Finally, put zip file on your secret HTTP location.

Export company logo, if you set custom company logo image url in settings, you can include it in your export services package. When user imports package, your company logo will automatically replace welcome page background.

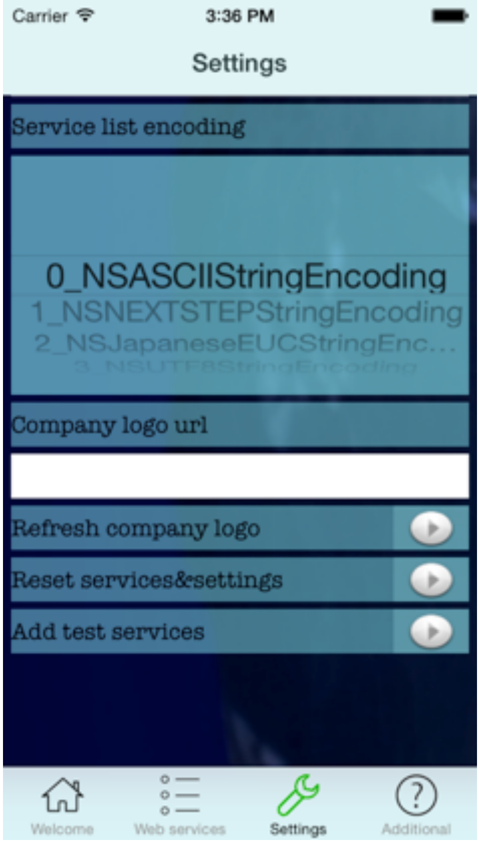
Import URL location, enter URL location of exported ZIP package previously described and tap on ▶. App will ask you for type of import:

1. Drop existing (deletes existing services and append imported ones)
2. Add to existing (preserves existing services and append imported ones)
3. Update existing services (if local service ID exists in import package, local service with that ID will be completely replaced, otherwise it will be added as new one)

Import passcode, password used for export with passcode action

Device ID, identifier which represents last 10 characters of unique device ID. You cannot change this value. This value will be sent to server with every web service call.

Send ID to web service, if switched on, every request to web service will contain device id. It is useful for security reasons, so you can isolate your web service, to serve only previously approved devices.



Settings

Service list encoding

0_NSASCIIStringEncoding
1_NSNEXTSTEPStringEncoding
2_NSJapaneseEUCStringEnc...
3_NSUTF8StringEncoding

Company logo url

Refresh company logo

Reset services&settings

Add test services

Service list encoding, if you have to specify some special chars in web service declaration. Otherwise, use NSUTF8 string encoding.

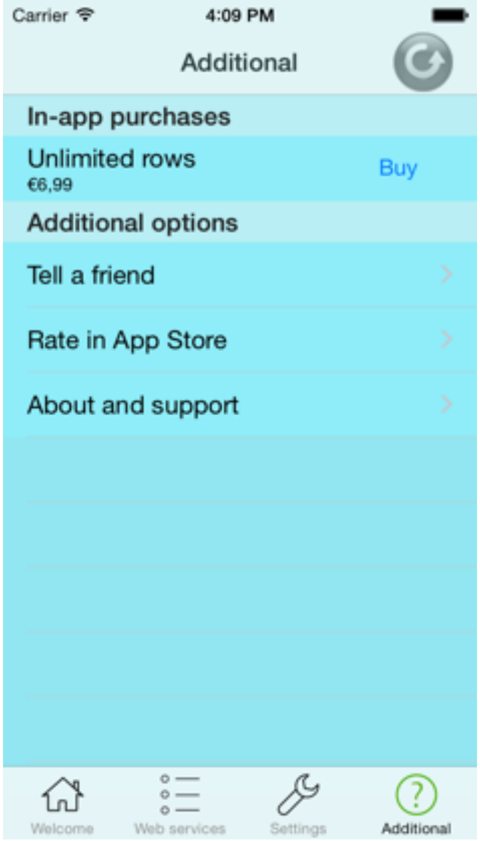
Company logo url, enter url path for company logo image. After that, when you jump to Welcome page, image will be downloaded and saved locally.

Refresh company logo, if you change url path, tap once on this button to reload logo image.

Reset services&settings, this action drops all services and resets settings to defaults.

Add test services, this action adds three test web services for presentation purposes.

Additional



Carrier 4:09 PM

Additional

In-app purchases

Unlimited rows
€6,99 Buy

Additional options

Tell a friend

Rate in App Store

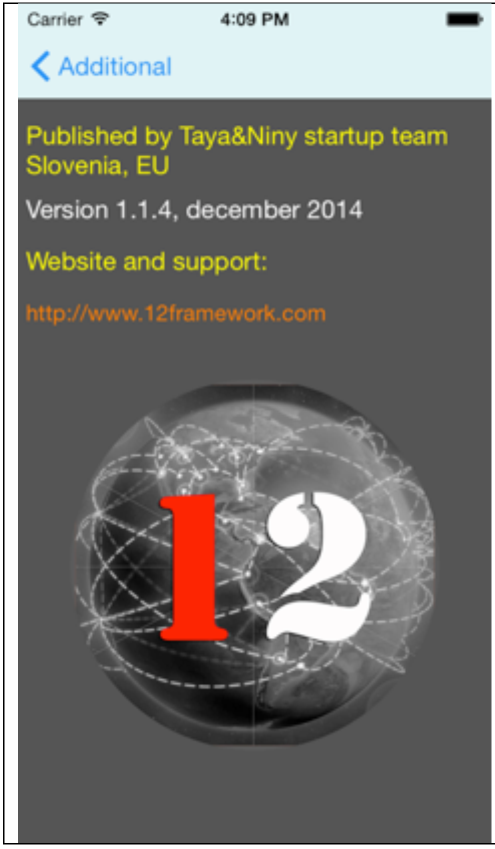
About and support

Additional page offers InApp purchase Unlimited rows.

Buying **removes 20 rows limit** of reading data and **enables adding new records.**

If you like this app, tell a friend or rate it in App Store.

Thank you.



Carrier 4:09 PM

< Additional

Published by Taya&Niny startup team
Slovenia, EU

Version 1.1.4, december 2014

Website and support:

http://www.12framework.com

About and support

This is about and support view. Tap on orange link to connect to 12framework website for further info or questions.

Thank you very much for using 12FrameWork!

Taya&Niny startup team